Coupled Real-Synthetic Domain Adaptation for Real-World Deep Depth Enhancement

Xiao Gu, Yao Guo, Fani Deligianni, and Guang-Zhong Yang*, *Fellow, IEEE*

***Abstract*—Advances in depth sensing technologies have allowed simultaneous acquisition of both color and depth data under different environments. However, most depth sensors have lower resolution than that of the associated color channels and such a mismatch can affect applications that require accurate depth recovery. Existing depth enhancement methods use simplistic noise models and cannot generalize well under real-world conditions. In this paper, a coupled real-synthetic domain adaptation method is proposed, which enables domain transfer between high-quality depth simulators and real depth camera information for super-resolution depth recovery. The method first enables the realistic degradation from synthetic images, and then enhances degraded depth data to high quality with a color-guided sub-network. The key advantage of the work is that it generalizes well to real-world datasets without further training or fine-tuning. Detailed quantitative and qualitative results are presented, and it is demonstrated that the proposed method achieves improved performance compared to previous methods fine-tuned on the specific datasets.**

***Index Terms*—Depth Enhancement, Real-World, Denoising, RGBD Sensor, Domain Adaptation, Deep Learning.**

I. Introduction

Accurate depth recovery is a pre-requisite for robotic navigation and manipulation [1], [2], surgical guidance [3], and human motion analysis [4]. Currently, sensing technologies supporting commercial depth cameras are mainly based on stereo correspondence, structured lighting, time-of-flight, or a combination of these techniques [5]. However, many factors can affect depth measurements, including noise, artifacts, biases, and interference [6]. Temporal stability can be affected by reflective materials and illumination sources. Furthermore, most depth sensors have lower resolution compared to that of the associated rgb images.

To overcome the above problems, extensive research has been devoted to depth quality enhancement, which involves either super-resolution or depth completion. A summary of existing methods is given in Section II. One basic pipeline for supervised depth enhancement is to add simulated noise onto ground truth depth images, and subsequently recover the high resolution features via deep learning. Due to the difficulty of acquiring high-quality data in the real world, the use of realistic simulators is common [7], [8]. These simulators are able to produce high-quality depth data and photo-realistic textured rgb images. To facilitate supervised depth enhancement, noise is simulated based on down-sampling and distance-based degradation. However, this operation does not adapt well to real-world applications, as the degradation patterns presented in real-world depth maps are much more complex. Thus, it is essential to bridge the gap between synthetic and real-world depth data for improved depth recovery [9], [10].

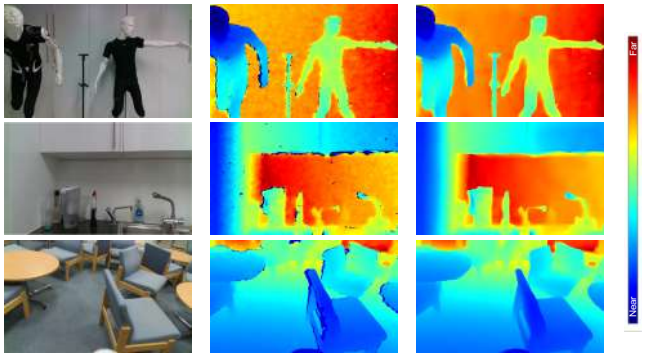


Fig. 1. Examples of real-world depth map enhancement. Color and depth images were taken by Intel RealSense D435. The third column shows the enhanced depth images by our proposed method. For better visualization, we map the original gray-scale depth image into a color image with 'jet' colormap. The same in the rest of the paper.

Thus far, several approaches have been developed to model noise in depth data and, subsequently, attempt to exploit this data for training of tasks related to real depth sensors [2]. Handa *et al.* [11] utilized a depth noise model to simulate noise from Kinect, only taking into account the distant dependent noise and geometry edge distortion. Planche *et al.* [12] predicted depth noise from 3D CAD models considering factors such as sensor noise and surface geometry. However, this method is difficult to be applied to indoor scenes as it only focuses on generating depth maps for single objects. Keller *et al.* [13] built a virtual time-of-flight sensor to capture depth images in virtual scenes. Whilst being possible to introduce physical characteristics into a simulator, it is hard to realistically reproduce noise related to unknown light sources, interference, and different reflective materials.

Hitherto, deep learning based methods have been used to generate realistic depth maps or model noise. Atapour-Abarghouei *et al.* [14] tried to predict holes (missing values in depth) from the corresponding rgb images. Similarly, Sweeney *et al.* [15] also proposed a supervised method for the prediction of hole locations from reconstructed complete

This work was supported by the Engineering and Physical Sciences Research Council (EPSRC) under Grant (EP/R026092/1). (Corresponding author: Guang-Zhong Yang.)

X. Gu and Y. Guo are with the Hamlyn Centre, Institute of Global Health Innovation, Imperial College London, London SW7 2AZ, UK (e-mail: {xiao.gu17, yao.guo}@imperial.ac.uk).

F. Deligianni is with the School of Computing Science, University of Glasgow, Glasgow G12 8QQ, UK (e-mail: fani.deligianni@glasgow.ac.uk).

G.-Z. Yang is with the Institute of Medical Robotics, Shanghai Jiao Tong University, Shanghai 200240, China (e-mail: gzyang@sjtu.edu.cn).

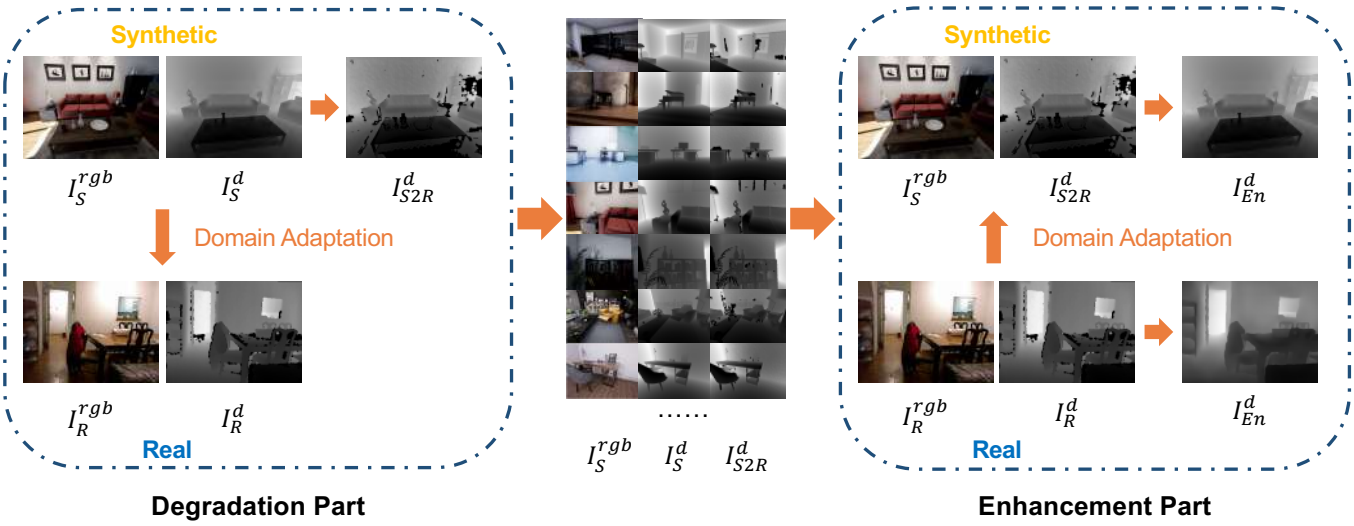


Fig. 2. Overview of the proposed depth enhancement framework. The whole framework consists of two parts, the degradation and the enhancement. The degradation part performs synthetic-to-real domain adaptation, converting high-quality synthetic depth images to realistic ones based on hole prediction and subsequent adversarial domain adaptation. Subsequently, the enhancement part converts the realistically degraded images back to high-quality synthetic images through a color-guided adversarial network. The enhancement part generalizes well on real-world depth images without fine-tuning.

depth maps in the real world, and the trained model can be used to simulate holes based on depth maps acquired from the simulator. However, these methods only simulate a single type of depth noise from a single modality. Thus, the predicted noises are simplistic and cannot model complex patterns in real-world settings. Shrivastava *et al.* [9] and He *et al.* [16] developed methods for realistic hand depth maps generation, enhancing the performance of hand pose estimation in real-world settings. However, they are constrained in generating specific objects without considering the surrounding environment.

In this paper, we present a novel framework for real-world depth enhancement based on coupled realistic degradation and enhancement via adversarial domain adaptation. Firstly, synthetic depth images generated from simulators are superimposed with holes (missing values in depth) predicted from corresponding rgb images. Subsequently, noise present in the real-world depth data is transferred to high-quality synthetic data based on adversarial training. This aims to optimize an adversarial loss of the noise model along with preserving the geometric structure of the synthetic depth data. This architecture results in realistic depth image degradation characteristics, such as geometric distortion and blurred boundaries. Subsequently, depth enhancement based on color-guided supervised training converts realistically degraded depth data to paired ground truth, simultaneously performing adaptation to high-quality synthetic depth domain by adversarial training. The proposed enhancement model can effectively enhance real-world depth data as the examples in Fig. 1 show. The workflow of the entire framework is presented in Fig. 2.

The novelty of our approach is two-fold:

Realistic depth noise modeling-we introduce a novel depth degradation method for synthetic depth images based on domain transfer of real-world data. Although deep learning has been applied to several domains that exploit depth information to address high-level problems of object detection, pose estimation and scene parsing, to the best of our knowledge, this is the first work that simulates realistic depth data in indoor scenes with deep neural networks.

Real-world depth enhancement pipeline-we propose a novel training pipeline for depth enhancement that targets real-world data even without ground truth, with the help of knowledge transfer between simulators and real world.

The remaining parts of the paper are structured as follows. Section II provides an overview of the existing works related to this paper. Section III demonstrates our proposed pipeline and methods in detail. The experiments and implementation details are presented in Section IV-B, followed by experimental results in Section V. Finally, we summarize the relative merits and potential pitfalls of our work in Section VI.

II. Related Works

A. Depth Enhancement

The enhancement of depth data from commercial depth sensors involves super-resolution (upsampling and denoising (non-hole noises)) and depth completion (hole filling). Conventionally, single view depth super-resolution relies on filtering techniques, such as bilateral filtering or regularization based optimization strategies. The former is based on the use of interpolation operations that facilitates the upsampling of the image, which can suffer from texture-copy related artifacts [17]. The latter optimization is achieved via Markov Random Field (MRF) by incorporating priors to ensure local depth smoothness [18]. Color-guided super-resolution has been used by both of these approaches to improve the performance [19], [20], [21], [22], [23]. These methods tend to involve complex optimization and thus high computational complexity.

a) Depth completion: Some depth enhancement approaches, targeting depth completion, involve filling holes/invalid values in depth maps [24]. Compared to the long-standing issue of color completion/inpainting, which has

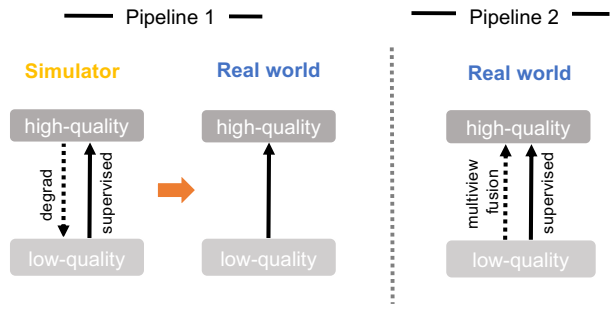


Fig. 3. Two general pipelines for deep learning based single depth enhancement method. The first pipeline firstly degrades the depth images by specific approaches and then performs training on the pair of {degraded (low-quality), ground truth (high-quality)}. The second one is based on the 'ground truth' acquired from the accumulation from nearby frames, and then trained on the pair of {raw (low-quality), ground truth (high-quality)}.

been improved significantly by deep learning approaches [25], methods for depth map completion are less well studied. The majority of these approaches are based on conventional methods [5]. For example, color information along with anisotropic diffusion principles [26] and low-rank matrix completion [27] have been used. Only a few deep learning methods exist that target depth completion [28], [14]. Zhang and Funkhouser [28] developed a network to predict the depth occlusion edges and surface normal vectors from rgb images, and subsequently applied a global optimization method to refine the depth map. However, for real-time applications, an end-to-end network is required that enhances depth resolution while it accurately fills depth holes and removes real-world depth artifacts.

b) Depth super-resolution: Recently, deep learning based methods have attracted considerable interest due to their success in the super-resolution application of rgb images. Hui *et al.* [7] proposed depth enhancement based on deep learning by adding a color-guided branch into their proposed single image super-resolution network [29]. Several color-guided deep network architectures [30], [31], [32] were proposed to improve upon Hui *et al.*'s work. Some researchers also explored solutions without color guidance [8], [33] to enhance depth quality. With these methods, however, areas with missing values may not be recovered well during depth enhancement. As mentioned earlier, the general pipeline (Pipeline 1 shown on the left of Fig. 3) for the training of these methods is based on synthetic ground truth and the corresponding down-sampled depth images with simplistic distance-dependent noise models. These techniques, however, can fail in real-world depth enhancement tasks similar to the real-world rgb super-resolution [10], [34].

Recently, Jeon and Lee [35] proposed a novel pipeline (the right side Pipeline 2 of Fig. 3) for depth enhancement, based on the ground truth depth data derived from multi-view reconstruction [36]. These ground truth images have relatively good quality and have been used extensively to benchmark depth enhancement techniques. However, occasional misalignment of the reconstructed meshes can cause large errors around edges. Using these data as the ground truth affects the convergence performance in end-to-end supervised training [35], [37].

B. Domain Adaptation

High-quality depth data of indoor scenes can be easily generated from simulators, while real-world depth data of degraded quality are acquired by rgbd cameras. Since no corresponding pairs exist, it is challenging to recover high-quality depth from real data in a supervised fashion. Therefore, domain adaptation is introduced [38], [39], [40]. This minimizes the difference of the data distributions between the target and source domains. As a result, the trained model in the source domains can be applied to unlabeled data in the target ones [41], [42], [43].

a) High-level adaptation: As concluded in [44], the visual domain adaptation algorithms can be roughly classified into two classes, feature-level (high-level) and pixel-level (low-level) approaches. High-level domain adaptation requires the extraction of invariant feature representations [45], [46], [47]. For example, in person re-identification tasks [46], a gradient reversal layer (GRL) was proposed to ensure the distribution similarity between the two domains. He *et al.* [47] proposed a progressive domain adaptation architecture for transferring object detection task from normal to foggy scenes, inserting GRL into several intermediate layers to align the embedding features. In [48], a reconstruction based approach was utilized. The distributions of embedding features were first minimized in terms of KL-Divergence and then applied to the regression problem, which demonstrated good performance in the target domain. In our case, to learn the noise characteristics in real-world depth images would be a low-level problem, since it involves pixel-level operations [10], [34]. Furthermore, realistic modeling of depth noise is important in applications involving depth data. They can be typically trained on simulators and then implemented in the real world, such as robot manipulation [2] and navigation [11].

b) Low-level adaptation: Pixel-level adaptation, also known as image translation, is essential for cross-domain transfer. This has been developed to address the lack of real-world labeled data. Recently, deep learning has been proposed to enhance the realism of synthetic data so that training on large synthetic datasets can be generalized to real-world data [9], [49], [39]. In Sim-GAN [9], the authors proposed a refiner network based on adversarial networks and reconstruction loss to convert images from synthetic to real while preserving geometric similarity. It can generate realistic hand images and this additionally generated dataset enhances the performance of hand pose estimation. James *et al.* [50] proposed a randomized-to-canonical adaptation network to transfer both randomized synthetic and real-world rgb images to a canonical simulated rgb domain, thus enabling transferring robotic grasping capacity learned in the randomized simulated domain to the real world.

Thus far, popular image to image translation (or style transfer) methods such as Cycle-GAN [51] and DualGAN [52] have been used for real-synthetic pixel-level domain adaptation. Without the need of paired data, cycle-consistency based architecture proposed by Cycle-GAN [51] can significantly improve unsupervised cross-domain image translation. This architecture has been successfully extended and implemented

in a range of applications. Bulat *et al.* [10] applied the cycle architecture to generate super-resolved facial images. Li *et al.* [53] proposed knowledge transfer between unpaired CT and X-ray images based on cycle-consistency loss, facilitating chest X-ray image decomposition. Jeong *et al.* [54] adopted the structure of Cycle-GAN and explored the use of cross-spectral correspondence between visible and infrared images in an unpaired setting. However, noise characteristics and data range representation are fundamentally different when using depth and rgb data. Furthermore, the so-called 'style transfer' in the rgb domain is limited. For example, when applied to the depth domain, it might adapt texture-based information from one domain to another or contaminate the depth range. Therefore, they may inadvertently remove existing objects or introduce additional structural artifacts.

C. Depth Estimation and Sparse Depth Reconstruction

Other related works to depth images, such as depth estimation and depth reconstruction from sparse samples, are summarized in this section. Although it includes ideas that have inspired our work, they are based on different principles and objectives as discussed below.

a) Depth estimation: Depth estimation refers to inferring depth information directly from monocular/stereo/multiview images or image sequences [49], [55], [56], [3]. Several benchmarks of depth recovery task for both indoor or outdoor scenes are available (e.g., NYU-V2[1], SUN-RGBD[2]). In fact, many state-of-the-art depth estimation methods are trained based on the rgb and ground truth depth pair in an end-to-end supervised fashion. However, the ground truth data is often recorded through rgbd cameras (e.g., NYU-V2), inevitably suffering from noise. The depth quality may be suitable for depth estimation, but it is insufficient for super-resolution depth enhancement.

The use of simulators for generating paired rgb and depth images have already been utilized for high-quality depth image estimation. Different to depth enhancement, depth estimation requires only rgb images as input. Thus the real-synthetic domain adaptation is limited to rgb images only. Mahmood *et al.* [57] proposed a low-level domain adaptation method for endoscopic depth estimation. The transformer network proposed in this work converted real medical images to synthetic-like ones so that the depth estimation model trained on synthetic rgb-depth pairs could work on real-world monocular endoscopy images. Zhao *et al.* [56] proposed a geometry-aware symmetric domain adaptation framework by leveraging the high-quality depth information from simulators and the epipolar geometry constraint of real-world stereo maps. This framework adopts coupled adversarial training for rgb style transfer, which is similar to Cycle-GAN, thus enabling rgb real-synthetic domain adaptation.

b) Sparse depth reconstruction: Sparse depth reconstruction/recovery refers to reconstructing a complete depth map from sparse samples. The public benchmark datasets for this task include CityScape[3] and KITTI[4]. Their raw depth datasets, which are recorded by LiDAR devices in outdoor scenes, are sparse and thus the noise patterns differ from those of rgbd cameras [28]. Although sparse depth reconstruction is relevant to our topic, it involves a different set of challenges.

[1] https://cs.nyu.edu/~silberman/datasets/nyu_depth_v2
[2] http://rgbd.cs.princeton.edu/
[3] https://www.cityscapes-dataset.com/
[4] http://www.cvlibs.net/datasets/kitti/

III. Methodology

A. Problem Formulation

The conventional color-guided depth enhancement algorithms can be formulated as follows,

$$\min \mathcal{L}^{En}(\mathcal{F}^{En}(I^d_{lq}, I^{rgb}), I^d_{hq}) \tag{1}$$

where $I^d_{lq}, I^d_{hq}, I^{rgb}$ represent the depth data of low-quality, depth data of high-quality, and corresponding rgb map, respectively.

As stated earlier, our aim is to enable realistic formation of low-quality depth information from high-quality simulated data, so we transform Eq. (1) into the following formulation to explicitly declare the realistic depth noise modeling

$$\min \mathcal{L}^{En}(\mathcal{F}^{En}(\underbrace{\mathcal{F}^{Degrad}(I^d_{gt}, I^{rgb})}_{I^d_{lq}}, I^{rgb}), I^d_{gt}) \tag{2}$$

where $\mathcal{F}^{Degrad}$ encodes the generation of I^d_{lq} that shares similar distribution with the real-world depth images and $\mathcal{F}^{En}$ encodes the step to recover the ground truth depth data from the degraded one.

B. Overview

Based on the above formulation, the proposed depth recovery framework consists of two parts, **depth degradation** and **depth enhancement**, as shown in Fig. 2. The degradation part aims to model the noise pattern and it is used to add realism to the ground truth synthetic data based on real data in an unsupervised fashion. The enhancement part generates high-quality depth maps from realistically degraded synthetic data based on supervised training. Our aim is to generalize the model for real-world depth enhancement.

It is worth noting that all the relevant annotations in Fig. 2 and the following contents are explained in the captions of Figs. 4 & 5.

C. Depth Degradation Model

1) Architecture: To simulate both missing and imprecise depth values, we apply two concatenated sub-networks as demonstrated in Fig. 4. The aim of the first sub-network is to predict a mask that encodes the regions of missing/invalid values from the rgb data [24]. The second part simulates other noise sources, such as unreliability near depth edges and depth discontinuity.

a) Hole prediction: For real depth cameras, missing values can be predicted based on featureless surfaces, specular reflections and illumination interference present in color images. Therefore, a hole prediction model G_{hole} is built to predict missing regions in depth I^d from the corresponding I^{rgb}, resembling a UNet like structure [14]. Meanwhile, a network G^{rgb}_{S2R} is introduced to perform rgb domain adaptation from Synthetic (S) data to Real-world (R) images. G^{rgb}_{S2R} is trained

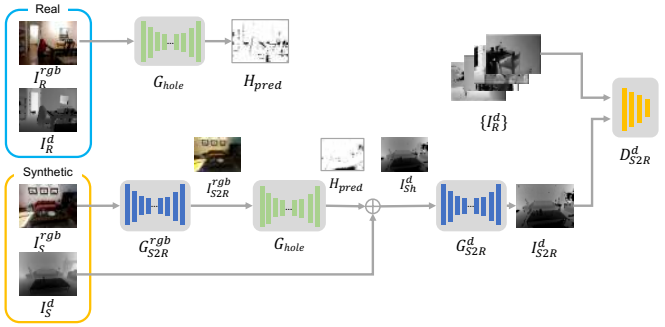


Fig. 4. Depth degradation architecture. I_S^{rgb} and I_S^d represent simulation-based rgb and depth images, respectively. H_{gt} represents the ground truth mask of missing depth values obtained from I_S^d. I_R^{rgb} and I_R^d represent realistic rgb and depth images captured by commercial depth sensors. G_{hole} refers to the network which predicts depth hole H_{pred} from corresponding rgb images, and subsequently generates the depth map added with hole I_{Sh}^d. G_{S2R}^{rgb} and G_{S2R}^d networks transfer rgb and depth images from synthesis to real-world separately. Lastly, D_{S2R}^d is a discriminator used to distinguish synthetic against realistic images.

based on the unsupervised approach Cycle-GAN[5] [51]. This adaptation aims to reduce the differences between realistic rgb images and those generated from simulators. The final depth hole prediction H_{pred} is generated from $G_{hole}(G_{S2R}^{rgb}(I_S^{rgb}))$.

b) Other degradation: In practice, it is challenging to predict realistic depth noise solely from its corresponding rgb data, since some geometric-based depth distortions as well as the hardware SNR characteristics may not be directly related to rgb maps. To alleviate this problem, an adversarial network G_{S2R}^d is used to enhance noise modeling based on real-world depth data. This method is inspired by recent work on adversarial training that combines both simulated and real images [9]. In this work, the generator part is composed of several ResNet Blocks [58], whereas the discriminator part is a patch discriminator [9], [51]. This architecture supersedes traditional discriminators as it models local features more precisely.

2) Loss Functions:

a) Hole prediction: Eq. (3) shows the objective functions based on the cross-entropy $\mathcal{L}_{CE}^{hole}$ and Jaccard distance $\mathcal{L}_J^{hole}$ between the estimated and ground truth hole maps, respectively [14]. This is a common method used in image segmentation.

$$\begin{cases} \mathcal{L}_{CE}^{hole} = \frac{1}{N}\sum(-H_{gt}log(H_{pred}) \\ \qquad\qquad -(1-H_{gt})log(1-H_{pred})) \\ \mathcal{L}_J^{hole} = 1 - \frac{H_{pred} \cap H_{gt}}{H_{pred} \cup H_{gt}} \end{cases} \tag{3}$$

From this hole map, the depth map I_S^d is degraded to the map superimposed with depth holes, I_{Sh}^d, by the formulation $I_S^d \odot (Sigmoid(H_{pred}) > 0.5)$, where $\odot$ represents the Hadamard's product.

[5]The Cycle-GAN model G_{S2R}^{rgb} & G_{R2S}^{rgb} is trained independently with code released on https://github.com/junyanz/pytorch-CycleGAN-and-pix2pix. In terms of the visual domain shift metrics FID (details in Section V-A2), (I_S^{rgb}, I_R^{rgb})= 0.6564; (I_{S2R}^{rgb}, I_R^{rgb})= 0.5062; (I_S^{rgb}, I_{R2S}^{rgb})= 0.5502; (I_{S2R}^{rgb}, I_{R2S}^{rgb})= 0.6032.

b) Other degradation: Sub-network G_{S2R}^d is trained based on the strategy proposed by Least Square GAN [59]. $\mathcal{L}_D^{S2R}$ enhances the discriminating ability between two domains, whereas $\mathcal{L}_G^{S2R}$ aligns the distribution of two domains. Both loss functions are optimized alternatively. Meanwhile, to preserve most of the original characteristics (e.g., depth range, depth geometry), a pixel-wise loss $\mathcal{L}_{Pixel}^{S2R}$ is used. In practice, the generator with the aim of $\mathcal{L}_D^{S2R}$ loss minimization is first trained to achieve faster convergence and avoid local optima.

$$\begin{cases} \mathcal{L}_D^{S2R} = \frac{1}{2}\mathbb{E}_{x\sim\mathbb{P}_{I_R^d}}[(D_{S2R}^d(x)-1]^2 \\ \qquad\qquad + \frac{1}{2}\mathbb{E}_{\hat{x}\sim\mathbb{P}_{I_{S2R}^d}}[(D_{S2R}^d(\hat{x})]^2 \\ \mathcal{L}_G^{S2R} = \frac{1}{2}\mathbb{E}_{\hat{x}\sim\mathbb{P}_{I_{S2R}^d}}[(D_{S2R}^d(\hat{x})-1]^2 \\ \mathcal{L}_{Pixel}^{S2R} = \left\|G_{S2R}^d(I_{Sh}^d) - I_{Sh}^d\right\|_2^2 \end{cases} \tag{4}$$

D. Depth Enhancement Model

1) Architecture: The depth enhancement model G_{En}^d, as shown in Fig. 5, utilizes both rgb and depth images and performs quality enhancement (simultaneous denoising and depth completion). We adopt a network structure similar to [63], which can estimate depth based on sparse depth samples.

2) Loss Functions: In addition to adversarial training loss, three loss functions are added to cater for the total loss function of the proposed network. These three terms are inspired by the penalty terms proposed for high-quality depth estimation in previous work [55].

$$\begin{cases} \mathcal{L}_{Pixel}^{En} = \left\|G_{En}(I_{S2R}^d) - I_S^d\right\|_2^2 \\ \mathcal{L}_{Grad}^{En} = \left\|\nabla_x(G_{En}(I_{S2R}^d) - I_S^d)\right\| \\ \qquad\qquad + \left\|\nabla_y(G_{En}(I_{S2R}^d) - I_S^d)\right\| \\ \mathcal{L}_{Norm}^{En} = \frac{1}{N}\sum(1 - \frac{\langle n_{pred}, n_{gt}\rangle}{\sqrt{\langle n_{pred}, n_{pred}\rangle}\sqrt{\langle n_{gt}, n_{gt}\rangle}}) \end{cases} \tag{5}$$

where $\mathcal{L}_{Pixel}^{En}$ represents the depth distance between the estimated and ground truth data, while $\mathcal{L}_{Grad}^{En}$ and $\mathcal{L}_{Norm}^{En}$ aim to

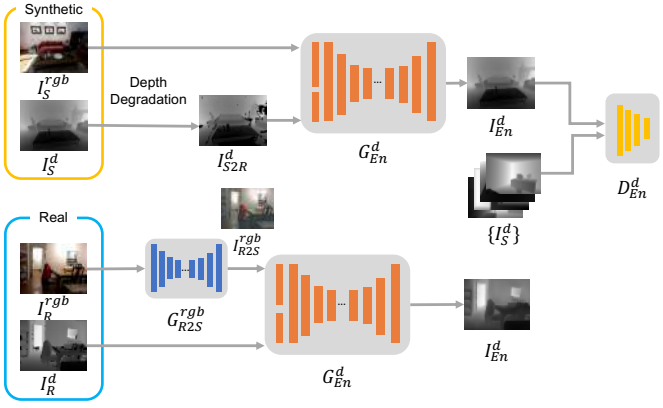


Fig. 5. The architecture of the enhancement network. I_S^{rgb} and I_S^d, I_R^{rgb} and I_R^d, represent rgb and depth images from simulators and real-world, respectively. G_{R2S}^{rgb} is the network that converts I_R^{rgb} to I_{R2S}^{rgb}, coupled with G_{S2R}^{rgb} in Cycle-GAN. The network G_{En}^d supplemented with the discriminator D_{En}^d is responsible for depth enhancement, and the output of which is I_{En}^d.

TABLE I
DATASET DESCRIPTION

Dataset	Syn/Real	Camera	Sensing Type	#	Size	Usage
UnrealCV [60]	Syn	-	-	2300	640×480	2000-training[§] and FID comparison, 300-testing
ScanNet [61]	Real	Structure*	structured light+stereo	3500	640×480	3000-training[♯] and FID comparison, 500-testing
RealSense	Real	RealSense D435[†]	structured light+stereo	132	1280×720	Evaluation in Sections V-A2 and V-D
NYU-V2 [62]	Real	Kinect V1[‡]	structured light	600	561×427	Evaluation in Sections V-A2 and V-D

* https://structure.io/ [†]https://www.intelrealsense.com/depth-camera-d435/ [‡]https://en.wikipedia.org/wiki/Kinect
[§,♯] The training data are also used for training G^{rgb}_{S2R}&G^{rgb}_{R2S}.

Fig. 6. Depth degradation results on UnrealCV. 1st-6th columns show synthetic rgb, synthetic high-quality depth, synthetic rgb with realism, predicted depth hole probability map, synthetic realistic depth map generated by proposed method and synthetic realistic depth map generated by Cycle-GAN.

minimize the total variance of depth errors and the distance between two normal maps n_{gt} and n_{pred}, respectively. For n_{gt} and n_{pred}, they are calculated as follows:

$$\begin{cases} n_{gt} = [-\nabla_x(I^d_S), -\nabla_y(I^d_S), 1]^T \\ n_{pred} = [-\nabla_x(G_{En}(I^d_{S2R})), -\nabla_y(G_{En}(I^d_{S2R})), 1]^T \end{cases} \quad (6)$$

Also, a discriminator similar to the one in Section III-C2 is used for adversarial training as follows:

$$\begin{cases} \mathcal{L}^{En}_D = \frac{1}{2}\mathbb{E}_{x\sim\mathbb{P}_{I^d_S}}[(D^d_{En}(x)-1]^2 \\ \qquad + \frac{1}{2}\mathbb{E}_{\hat{x}\sim\mathbb{P}_{I^d_{En}}}[(D^d_{En}(\hat{x})]^2 \\ \mathcal{L}^{En}_G = \frac{1}{2}\mathbb{E}_{\hat{x}\sim\mathbb{P}_{I^d_{En}}}[(D^d_{En}(\hat{x})-1]^2 \end{cases} \quad (7)$$

IV. EXPERIMENTS AND MATERIALS

A. Materials

To train and validate the proposed model, we used data as listed in Table I. High-quality synthetic data were obtained with UnrealCV simulator (2300 samples - 300 for testing and 2000 for training) [60]. Real-world data was extracted from ScanNet (3500 raw images - 3000 for training only in degradation and 500 for testing) [61] along with corresponding ground truth [28][6], which has been used in [28] and [35] for depth enhancement tasks. Real-world data have also been extracted from NYU-V2 dataset (Kinect V1) [62] as well as acquired indoor data based on a RealSense sensor (D435).

[6]To avoid large misalignment during rendering in ScanNet dataset, we applied Structural Similarity (SSIM) index to do pre-filtering, where {raw, ground truth} pairs of small SSIM values are filtered out.

B. Experimental Implementation

The proposed models[7] were implemented with Pytorch and trained on NVIDIA Titan XP with random initialization (except for the ResNet Block in G^d_{En} which was initialized with pre-trained ResNet-34 [58]). The depth degradation and enhancement parts were trained progressively to allow for a transparent evaluation of each part of the proposed framework. This is similar to the settings used in [64], [65]. For all training, an adaptive learning rate optimization approach, Adam was used, with learning rate λ initialized as $1e^{-}2$ and divided by 2 every $100k$ iterations after the first $300k$ iterations until reaching $800k$. The images were online randomly cropped to 256×256 during training, together with on-the-fly data augmentation like vertical or horizontal flipping to avoid overfitting. Besides, G^{rgb}_{S2R} and G^{rgb}_{R2S} were trained using official Cycle-GAN with the default settings.

V. RESULTS

A. Depth Degradation

1) Qualitative results: Fig. 6 demonstrates the results of realistic depth degradation on UnrealCV. The simulated noise model results from reflective surfaces, geometry boundaries, small objects, illumination interference and other factors. We compared our results with a Cycle-GAN architecture for depth degradation learning, the generator and discriminator of which are the same as in Section III-C2b. As shown in the last row in Fig. 6, the degraded results by Cycle-GAN have introduced

[7]Project details are available on https://xiaogu.site/RDE

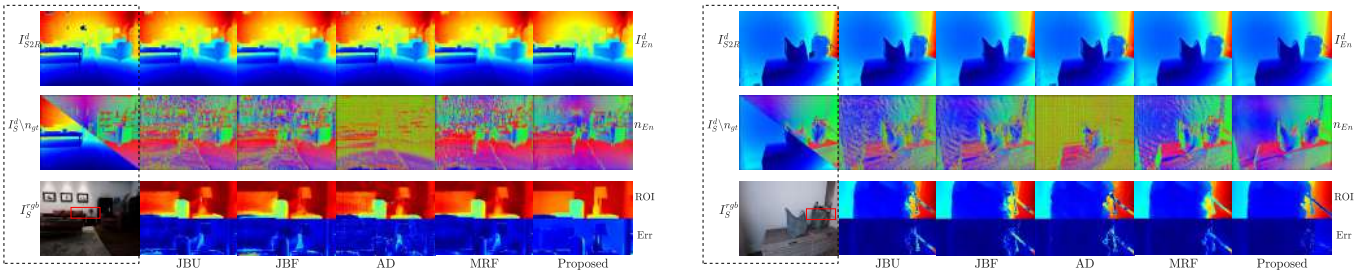


Fig. 7. Qualitative results of UnrealCV. The 1st column shows degraded synthetic depth I^d_{S2R} (top row), original synthetic depth I^d_S and normal map n_{gt} (mid row), and color I^{rgb}_S (bottom row). The 2nd-5th columns show the results of different approaches as listed in Table III, which include enhanced depth map I^d_{En} (top row), corresponding normal map n_{En} (mid tow), and close-up of region of interests (bottom row-top subrow: depth map; bottom row-bottom subrow: depth error map). Each group is normalized for visualization.

TABLE II
DOMAIN SHIFT (FID METRIC) BETWEEN DIFFERENT DATASETS

	RealSense	NYU-V2	UnrealCV	$I^d_{S2R}{}^*$	$I^d_{S2R}{}^\dagger$
ScanNet	1.1522	0.8076	1.3188	0.6878	0.9324
RealSense	-	0.9776	1.6005	1.0407	1.0846
NYU-V2	-	-	1.8890	1.2604	0.8275

* UnrealCV degraded to real-world with the proposed method.
† UnrealCV degraded to real-world with Cycle-GAN.

texture-like artifacts, thus contaminating the original depth range to a severe extent. This is because Cycle-GAN performs style transfer without structure preserving. In other words, our depth noise modeling strategy in Section V-A for depth degradation provides a better method for modeling realistic depth noise.

2) Domain shift comparison: To further evaluate the realism of our data, a domain distance metric Frechet-Inception-Distance (FID[8] [66]) was used, similar to [10]. We triplicated the depth channel to feed it into the Inception network and then statistical analysis was applied based on the embedding feature layers to test the domain distribution similarity of the two groups. Table II presents the FID measure between different sets of depth maps, including the datasets listed in Table I as well as the degraded depth data from UnrealCV based on our proposed method and Cycle-GAN. Smaller FID value in Table II indicates a smaller distance across domains. The FID, to some degree, represents the visual similarity across two groups, and it has already been used to evaluate rgb image super-resolution performance in the absence of ground truth data in [10]. However, in depth data, we should also consider other ways to evaluate the structure and smoothness of the maps. As observed in the last row of Table II, Cycle-GAN shows slightly better results than ours in terms of FID measure. However, the structure and range are largely contaminated with artifacts. We have further results/discussions in Sections V-C3 &V-C4 to highlight the realism of our degradation methods.

B. Depth Enhancement

We have quantitatively evaluated the depth enhancement process with 300 synthetic UnrealCV data with realistic noise, generated by the proposed depth degradation model. To

[8]https://github.com/mseitzer/pytorch-fid.

TABLE III
QUANTITATIVE RESULTS ON SYNTHETIC DATASET

Method	UnrealCV [60]					
	RMSE↓	RMSE$^{\phi}$ ↓	RMSE$^{\overline{\phi}}$ ↓	SSIM↑	PSNR↑	FID↓
JBU [17]	0.1129	0.4472	0.0729	0.9549	21.4093	0.6420
JBF [67]	0.1025	0.3580	0.0739	0.9581	22.1794	0.5340
AD [26]	0.1565	0.6497	0.1016	0.8502	18.1911	1.0465
MRF [18]	0.1007	0.3572	0.0727	0.9636	22.3032	0.6023
Proposed	**0.0907**	**0.2205**	**0.0704**	**0.9817**	**22.6439**	**0.3490**

demonstrate the effectiveness of the proposed model for depth enhancement tasks, we report the comparison results by Joint Bilateral Filtering(JBF) [67], Joint Bilateral Upsampling(JBU, upsampling factor=1) [17], Anisotropic Diffusion (AD) [26], and Markov Random Field (MRF) [18], on degraded UnrealCV in Table III. The parameters of these baseline methods were fine-tuned on UnrealCV dataset based on the training subsets mentioned in Table I. Fig. 7 shows the comparative results for enhancing an image from UnrealCV dataset.

In terms of metrics[9], the Root Mean Square Error (RMSE) and Structural Similarity (SSIM) index, Peak Signal-to-Noise Ratio (PSNR), and FID (distance from UnrealCV training set) were calculated. To illustrate the enhancement performance into hole inpainting and denoising respectively, we also report RMSE $^{\phi}$ (RMSE of areas where are holes in raw depth) and RMSE $^{\overline{\phi}}$ (RMSE of areas where are not holes in raw depth map), respectively. As shown in Table III, our method achieves superior performance against other methods, and for the visual quality metric FID, it correlates well with other metrics.

It should be noted that the depth enhancement model was trained on the synthetic data, which share similar rather than the same noise distribution with the real-world. The superior performance of the proposed method on synthetic data may be partly explained by information leak or overfitting. Nevertheless, the proposed method generalizes well on previous unseen real-world depth data. This is described below to emphasize that the degradation network learns seamlessly noise characteristics from real-world depth data.

C. Real-world Data Evaluation

1) Materials, Comparative Methods and Metrics Used: Inspired by the reconstruction based depth enhancement or

[9]The depth maps are compared in the unit of meter.

TABLE IV
QUANTITATIVE RESULTS ON REAL-WORLD DATASET

Method	ScanNet [61]					
	RMSE↓	RMSE$^{\phi}$ ↓	RMSE$^{\overline{\phi}}$ ↓	SSIM↑	PSNR↑	FID↓
JBU [17]	0.1663	0.3724	0.1304	0.9350	17.7865	0.6125
JBF [67]	0.1804	0.4327	**0.1303**	0.9322	18.4775	0.6824
AD [26]	0.1562	0.4436	0.1354	0.9306	18.8562	0.8257
MRF [18]	0.1567	**0.3015**	0.1396	**0.9397**	18.3025	0.7012
Refiner [35]	0.1831	0.4076	0.1469	0.9110	16.8948	0.6760
DC [28]	0.1564	0.3069	0.1364	0.9352	18.8591	0.8518
Cycle* [51]	0.2479	0.4992	0.2168	0.8798	13.9847	0.5732
Cycle† [51]	0.2322	0.7273	0.1480	0.8839	15.1609	0.6695
$\widehat{G}^{d}_{En}$ ‡	0.1838	0.4107	0.1470	0.9188	17.3607	0.7463
$\widehat{G}^{d}_{En}$ §	0.1648	0.3682	0.1310	0.9368	18.6400	0.7136
Proposed	**0.1511**	0.3032	0.1332	0.9310	18.5234	**0.5411**
Proposed¶	0.1543	0.3217	0.1326	0.9347	**18.9693**	0.5422

* Cycle-GAN with depth as input
† G^{d}_{En} trained with Cycle-GAN's degradation results (depth as input)
‡ $\widehat{G}^{d}_{En}$ trained only with 2000 ScanNet training pairs
§ $\widehat{G}^{d}_{En}$ trained only with 3000 ScanNet training pairs
¶ Direct end-to-end training for our proposed method

completion methods [35], [28], the rendered depth image from reconstructed meshes were utilized as the ground truth. To eliminate as many possible errors in these ground truth data due to misalignment, we have selected 500 ScanNet {raw, ground truth} pairs with high SSIM for testing. These are listed in Table I.

We compared our model with the traditional approaches (JBU, JBF, AD, MRF) listed in Section V-B, as well as state-of-the-art deep neural network methods for depth denoising (Refiner [35]) or depth completion (DC [28]). The implementation details of our method are included in Section IV-B. For traditional approaches, the parameters of JBU, JBF, AD, and MRF were fine-tuned based on the training set in ScanNet dataset mentioned in Table I. Pretrained models on ScanNet of Refiner[10] and DC[11] and their training details are also available online. The details of other methods compared are specified in each of their corresponding sections.

For quantitative validation, the same metrics[12], RMSE, RMSE$^{\phi}$, RMSE$^{\overline{\phi}}$, SSIM, PSNR, and FID are listed in Table IV, supplemented with comparative qualitative results in Figs. 8&9.

2) Overall results: Quantitative results are reported in Table IV and qualitative results including the enhanced depth, normal map, and close-up regions are shown in Fig. 8. The results demonstrate that the proposed method has achieved better, or at least similar performance compared to the state-of-the-art methods. The FID metric, which correlates well with other measures (those compared against high-quality synthetic data) in Table III, also highlights the superior performance of the proposed method in Table IV.

As listed in Table IV, the proposed model outperforms state-of-the-art deep neural networks on ScanNet. This also highlights the effectiveness of the proposed degradation model. Recovered depth and normal maps for each method are shown in Fig. 8 and demonstrate the superior performance of the proposed method both globally and in the highlighted regions. Normal maps reveal subtle fluctuations and a lack of smoothness that they are difficult to observe in the simple depth maps, alone. For example, the estimated high-resolution depth maps in AD [26] and DC [28] seem of high visual quality and they result in low RMSE. However, normal maps reveal profound noise and fluctuations.

3) Comparison with Cycle-GAN: Extended from Section V-A, two Cycle-GAN [51] based methods were run to compare pixel-level domain adaptation methods, Cycle* and Cycle†. The results are shown in Table IV and Figs. 8&9. In Section V-A, we set realistic depth and high-quality synthetic depth maps as input to perform unsupervised coupled domain adaptation based on the default settings in Cycle-GAN. The low-quality domain adaptation degraded examples are shown in the last column of Fig. 6. The high-quality domain adaptation performs depth enhancement and results are shown in Table IV and Figs. 8&9 marked as Cycle*. It can be observed that the Cycle-GAN can improve the visual quality significantly by transferring the characteristics from the simulator. However, it cannot keep the geometric characteristics well and may introduce distance bias.

As the first unsupervised method does not involve rgb for guidance, we extracted the degraded depth by Cycle-GAN and applied our proposed color-guided depth enhancement part for training based on the experimental details in Section IV-B. Since the Cycle-GAN for depth degradation can introduce large distance bias (Fig. 6), the degraded depth is firstly normalized to the original one. The enhancement results of the variant shown in Figs. 8&9 and Table IV marked as Cycle† are inferior to our method, demonstrating that the learned noise from the Cycle-GAN (the first method) does not match well with the real-world noises. In other words, Cycle-GAN is not suitable for aligning the source and target low-quality depth domain.

4) Comparison with the other pipeline: On the other hand, methods based on training alone on real-world {raw, ground truth} pairs (Pipeline 2 in Fig. 3) suffer from the fact that misalignment introduces errors around the edges. The method Refiner [35], was trained on real-world pairs in ScanNet without color guidance, which slightly distorted depth geometry. Furthermore, it cannot cope well with invalid depth values/holes. Similarly, we isolated the depth enhancement part of our proposed method, G^{d}_{En}, and trained it on real-world {raw, ground truth} ScanNet dataset directly without adversarial training, based on the training settings in Section IV-B. This network is an improved version of the Refiner network [35], since it incorporates color guidance. Results marked as $\widehat{G}^{d}_{En}$ ‡ and $\widehat{G}^{d}_{En}$ § in Table IV reflect the performance of G^{d}_{En} that was trained only with 2000 and 3000 real-world training pairs, respectively. Quantitative results are based on the same 500 testing set. Even with an increased sample size of 3000 for supervised training directly on real-world {raw, ground truth}, these approaches are inferior to our pipeline because they are affected by subtle misalignments introduced from reconstruction. In other words, Table IV shows that our training based

[10] https://github.com/JunhoJeon/depth_refine_reconstruct
[11] https://github.com/yindaz/DeepCompletionRelease
[12] Some small holes in ground truth depth due to inevitable occlusions are ignored during metrics calculation. The FID is calculated by comparison against the 2000 UnrealCV training depth maps in degradation part (See Table I)

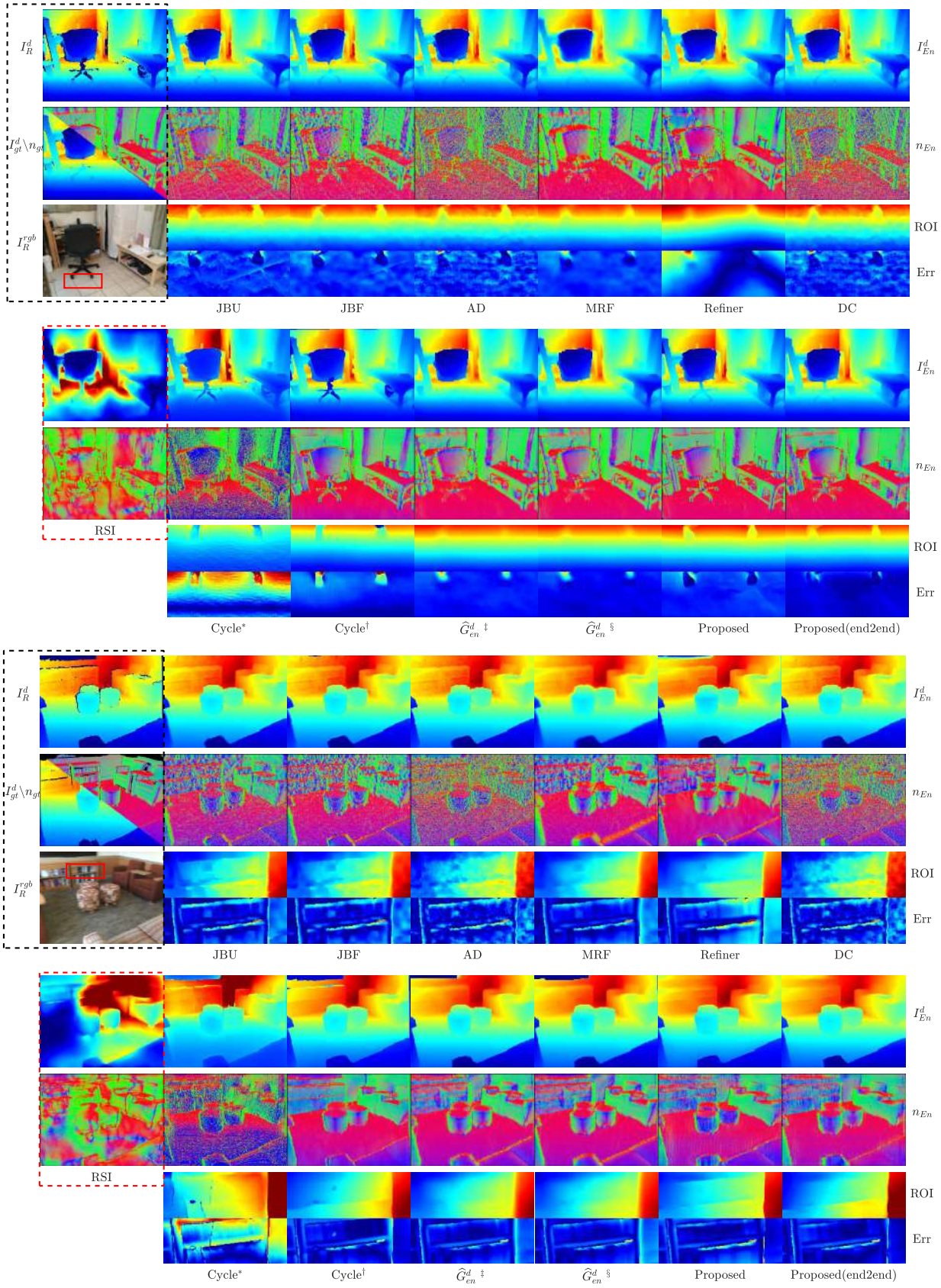


Fig. 8. Qualitative results of ScanNet. The upper part and the lower part show two examples. In each part, the 1st column (1st-3rd row) shows raw input I_R^d (1st row), ground truth depth I_{gt}^d and normal map n_{gt} (2nd row), and color I_R^{rgb} (3rd row). The 2nd-7th (1st-3rd row) and 3rd-7th (4th-6th row) columns list the results of different approaches listed in Section V-C and Table IV, which include enhanced depth map I_{En}^d (top row), corresponding normal map n_{En} (mid tow), and close-up of region of interests (bottom row-top subrow: depth map; bottom row-bottom subrow: depth error map). The left bottom corner (red rectangle) shows the result of depth estimation method RSI [55]. Each group is normalized for visualization and fair comparison.

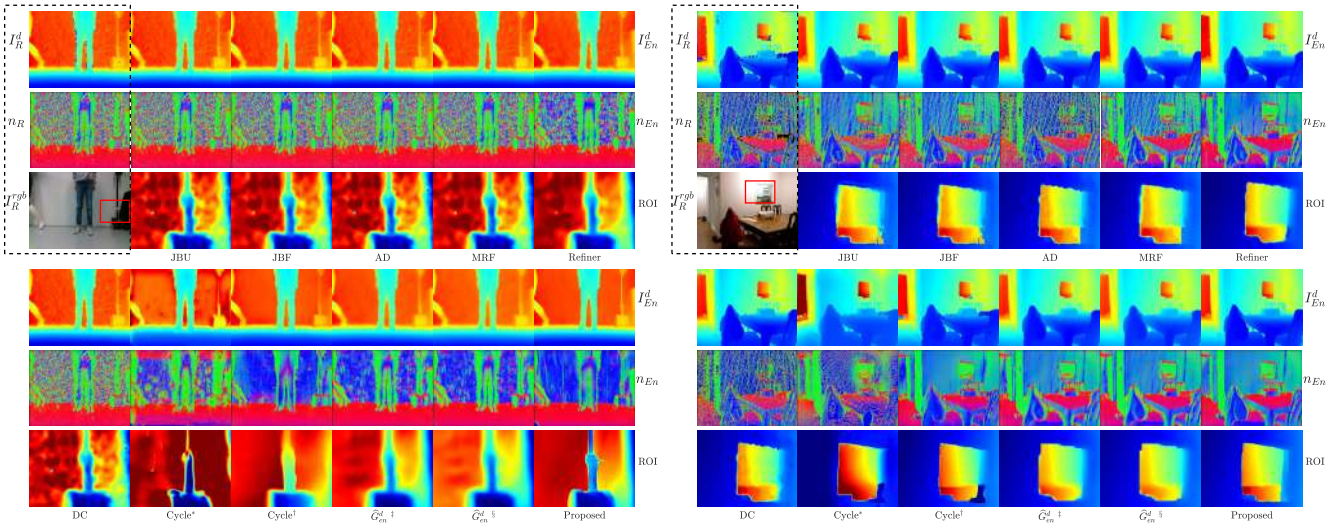


Fig. 9. Qualitative results of cross-dataset performance on real-world data. Left: RealSense D435; Right: NYU-V2 Dataset (Kinect V1). In each part, the 1st column (1st-3rd row) shows raw input I_R^d (1st row), raw normal map n_R (2nd row), and color I_R^{rgb} (3rd row). The 2nd-6th (1st-3rd row) and 1st-6th (4th-6th row) columns show the results of different approaches listed in Section V-C and Table IV, which include enhanced depth map I_{En}^d (top row), corresponding normal map n_{En} (mid tow), and close-up of region of interests (bottom row). Each group is normalized for visualization and fair comparison.

on synthetic {degraded, ground truth} pairs is sufficient to develop an enhancement approach that generalizes well to previously 'unseen' real data. The results also indicate the ability of our depth degradation part to model depth noise realistically.

5) Comparison of training strategy: To reduce the memory and computation cost, we applied a progressive training strategy as clarified in Section IV-B. This strategy does not hinder the novelty as well as the performance of our proposed model. We have provided both the qualitative and quantitative validation for each subnetwork, as well as the overall performance on real-world datasets. To further compare both, results of a direct end-to-end training version are also provided in Fig. 8 and Table IV. The results show similar performance compared to the original training strategy.

6) Comparison with depth estimation results: Results from the state-of-the-art depth estimation method RSI [55] are also compared in Fig. 8. As shown in Fig. 8, the relative depth range cannot be exactly matched without the prior knowledge of raw depth.

7) Comparison of run time performance: The running time of depth enhancement methods (Image resolution:640x480; GPU: Titan XP; CPU: Intel i9-7940X) are listed in Table V. Most conventional, non-deep methods, such as AD, MRF and JBU, are based on iterative optimization strategies and thus their convergence is slow.

Our method shows much better results than that of Cycle, although it takes longer time. Overall, the computational performance of our method is much more efficient compared to other deep learning based methods. It could potentially run in real-time with high-end GPUs along with optimized deep learning architectures.

Similarly to DC, which consists of occlusion and normal estimation sub-networks and a global optimization stage, our framework involves multiple sub-networks during training. However, in the testing phase, our method is an end-to-end architecture and, unlike DC, does not require computationally complex optimization strategies.

TABLE V
RUNNING TIME OF DEPTH ENHANCEMENT METHODS

Method	Platform	Time (second)
JBU [17]	Matlab(C)	78.8860
JBF [67]	Matlab(C)	36.8540
AD [26]	Matlab(C)	14.5093
MRF [18]	Matlab(C)	1.9086
Cycle [51]	Pytorch(G)	0.0029
Refiner [35]	Pytorch(G)	0.0967
DC [28]	Torch(G)+Matlab(C)	0.6901+25.4189
Proposed	Pytorch(G)	0.0288

D. Cross-dataset Performance

We further applied the fully trained model on the other two datasets, NYU-V2 and our own dataset recorded by RealSense, to evaluate the cross-dataset generalization performance. We directly applied our model and those methods used for comparison pre-trained on ScanNet in Section V-C without further training on these two datasets. This comparison enables the validation of cross-dataset generalization performance. Since there is no ground truth data for NYU-V2 and RealSense datasets, qualitative results are shown in Fig. 9. It can be observed that even though the depth degradation part was not trained on these two datasets, our method still achieves good visual quality on both depth and normal maps. To be more specific, for the RealSense dataset, the left side of Fig. 9 demonstrates that our method outperforms others that fail in handling the low SNR present in RealSense. This becomes apparent in depth enhancement of relatively small objects. Similarly, for the NYU-V2 dataset, our method shows well-defined and sharp boundaries. The outstanding performance highlights the good generalization ability of our method.

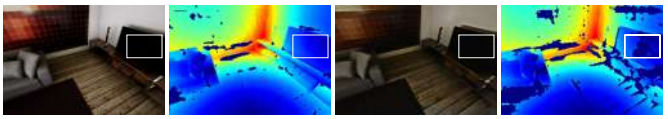


Fig. 10. Qualitative results of the ablation of G_{S2R}^{rgb} in degradation part. The 1st and 3rd columns represent synthetic and realistic rgb maps, respectively. The 2nd and 4th columns represent the degraded depth maps with and without G_{S2R}^{rgb}. Featureless areas have a higher probability of missing values after rgb domain adaptation.

TABLE VI
ABLATION STUDY ON SCANNET

Variants	ScanNet [61] RMSE↓	SSIM↑	PSNR↑	FID↓
Network				
w/o G_{S2R}^{rgb}	0.2474	0.8734	14.8219	0.6522
w/o G_{hole}	0.2795	0.8530	13.7214	0.6898
w/o G_{S2R}^{d}	0.1834	0.8877	17.8805	0.7932
w/o G_{R2S}^{rgb}	0.1564	0.9321	18.7026	0.5688
$G_{R2S}^{rgb} \rightarrow G_{S2R}^{rgb}$	0.1906	0.9153	17.0557	0.7669
Loss (w/o D_{En}^{d})				
w/ P	0.1953	0.9082	16.6723	0.6273
w/ $P+G$	0.1842	0.9156	17.0116	0.6344
w/ $P+G+N$	0.1636	0.9282	18.2898	0.5915

E. Ablation Study

To evaluate the effectiveness of different parts of the network, we trained the network based on several variants and show the results on ScanNet data [61] in Table VI.

1) Network Architecture:

a) Removal of each functional unit: In this part, the network architecture was changed by deliberately removing different parts of the network while keeping the same training procedure for other parts. They are G_{S2R}^{rgb} (synthetic to real-world rgb), G_{hole} (depth hole prediction), G_{S2R}^{d}&D_{S2R}^{d} (further depth degradation), and G_{R2S}^{rgb} (real-world to synthetic rgb). As shown in Table VI, each part of our architecture plays a role in real-world depth enhancement.

b) RGB domain adaptation: The simultaneous rgb (G_{S2R}^{rgb} & G_{R2S}^{rgb}) and depth domain adaptation is to mitigate both rgb and depth domain shift between real world and simulator. For depth, the domain margin mainly lies on the image quality, where real-world depths suffer from noises. For rgb images, the style difference including the lighting conditions would affect the depth holes predicted by the rgbd images in the depth degradation part. To apply the real-world trained hole prediction model on the synthetic domain, G_{R2S}^{rgb} is used to add realism by S2R translation. The qualitative results are shown in Fig. 10. It can be viewed that after domain adaptation, featureless areas such as the TV screen have a higher probability of holes occurring. This resembles real-world conditions better, and thus it enables training of a more realistic model that results in better depth-enhancement performance. The quantitative results of ablation of G_{R2S}^{rgb} are shown in Table VI marked as w/o G_{R2S}^{rgb}.

On the other hand, in the depth enhancement part, the rgb image resolution influences critically the domain shift process and subsequently the recovered depth quality, as shown in Figs. 11&12. Theoretically, a higher-quality reference rgb map

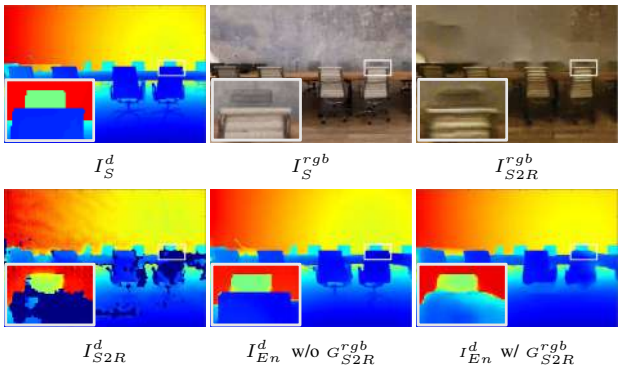


Fig. 11. Qualitative results of the variant of depth enhancement part on synthetic data. From left to right, the first row displays synthetic ground truth depth, synthetic rgb I_S^{rgb}, and synthetic rgb after rgb domain adaptation I_{S2R}^{rgb}. The second row displays synthetic degraded depth, enhanced depth with input as I_S^{rgb}, and enhanced depth with input as I_{S2R}^{rgb}.

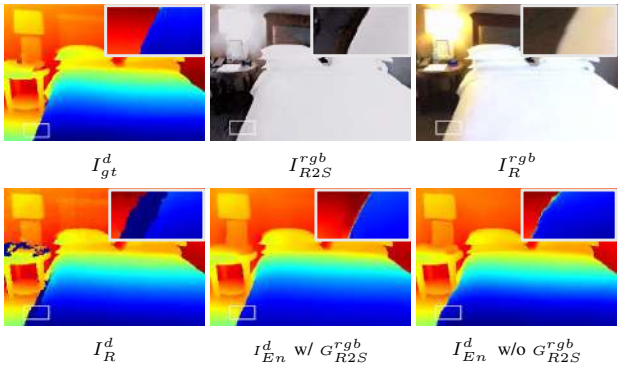


Fig. 12. Qualitative results of the variant of depth enhancement part on real-world data. From left to right, the first row displays real-world ground truth depth, rgb map after domain adaptation I_{R2S}^{rgb}, and real-world rgb I_R^{rgb}. The second row displays real-world raw depth, enhanced depth with input as I_{R2S}^{rgb}, and enhanced depth with input as I_R^{rgb}.

would lead to a higher-quality depth map. To validate it, the experiment with a variant of enhancement network was run, where during training or testing of synthetic images, the synthetic rgb firstly goes through pre-trained G_{S2R}^{rgb} before being fed into G_{En}^{d} while in the testing of real images, the real rgb directly goes to G_{En}^{d}. The results in both simulator and real world are shown in Figs. 11&12. The qualitative results are consistent with the quantitative results indexed as $G_{R2S}^{rgb} \rightarrow G_{S2R}^{rgb}$ in Table VI, which shows that the variant network performs worse than our proposed framework. Apparently, a lower-quality rgb input forces the single network to recover high-quality rgb and depth simultaneously, thus adding to the task complexity and compromising the recovery of higher-quality depth map. This is unnecessary, since current rgbd cameras provide high-quality of rgb images regardless of the depth quality. Therefore, in our settings, we decompose the first task, the high-quality rgb recovery, to the G_{R2S}^{rgb} for practical applications.

However, it should be noted that this paper is not focused on rgb domain adaptation. Simply, we adopted the state-of-the-art unsupervised Cycle-GAN directly to perform rgb domain adaptation.

2) Loss Function Components: In this part, different loss components of the enhancement part were combined. We

removed adversarial training loss D^d_{En}, and show the results trained on different combinations of $\mathcal{L}^{En}_{Pixel}$ (Pixel Loss: P), $\mathcal{L}^{En}_{Grad}$ (Gradient Loss: G), and $\mathcal{L}^{En}_{Norm}$ (Norm Loss: N). As shown in Table VI, each component of loss functions contributes to the performance of real-world depth enhancement.

VI. Conclusions

In this paper, we have introduced a novel color-guided method for real-world depth enhancement based on coupled real-synthetic domain adaptation. This method consists of two parts, namely the degradation and the enhancement part. Unlike previous methods, we do not directly rely on supervised training based on synthetic ground truth data and simplistically simulated noise. Instead, inspired by adversarial architectures, we exploit domain transfer to add realistical noise in high-quality synthetic data. Both qualitative and quantitative results demonstrate that the proposed method achieves realistic complex depth degradation.

In the proposed method, the enhancement part exploits color-guided, supervised learning based on pairs of realistically degraded synthetic data alone, together with minimizing adversarial training loss. In this way, the enhancement network structure achieves simultaneously depth completion and depth denoising that generalizes well to real-world depth data without ad hoc tuning or training. We tested the proposed enhancement model with both photo-realistic simulated datasets and real-world images, demonstrating the enhanced performance of the method proposed.

Further work should aim to take into account the temporal information to ensure the consistent high-quality recovery across time, as well as applying the proposed method to enhance the performance of related practical applications.

Appendix

A. Architecture Details

Network architecture details of G_{hole}, G^d_{En}, D^d_{S2R}/D^d_{En} are detailed in Table VII, VIII & IX. Each convolution/deconvolution layer is concatenated by BatchNorm and LeakyReLU layers except for the final output layer. Besides, G^d_{S2R} adopts the default settings of ResnetGenerator realeased in CycleGAN (https://github.com/junyanz/pytorch-CycleGAN-and-pix2pix), on which our networks related to rgb domain adaptation are also based.

TABLE VII
Architecture of G_{hole}

Name	K	S	Ch I/O	Res I/O	Input
Conv1	4	2	3/64	256/128	I^{rgb}
Conv2	4	2	64/128	128/64	Conv1
Conv3	4	2	128/256	64/32	Conv2
Conv4	4	2	256/512	32/16	Conv3
Conv5	4	2	512/512	16/8	Conv4
DeConv1	4	2	512/512	8/16	Conv5
DeConv2	4	2	1024/256	16/32	Conv4⊕DeConv1
DeConv3	4	2	512/128	32/64	Conv3⊕DeConv2
DeConv4	4	2	256/64	64/128	Conv2⊕DeConv3
DeConv5	4	2	128/64	128/256	Conv1⊕DeConv4
Conv6	4	2	64/1	256/256	DeConv5

* K: kernel size; S: stride; Ch: channel; Res: resolution.

TABLE VIII
Architecture of G^d_{En}

Name	K	S	Ch I/O	Res I/O	Input
Conv_rgb	3	1	3/48	256/256	I^{rgb}
Conv_d	3	1	1/48	256/256	I^{d}
Conv1	Res34-B1		96/64	256/256	Conv_rgb⊕Conv_d
Conv2	Res34-B2		64/128	256/128	Conv1
Conv3	Res34-B3		128/256	128 /64	Conv2
Conv4	Res34-B4		256/512	64/32	Conv3
Conv5	3	2	512/512	32/16	Conv4
DeConv1	3	2	512/256	16/32	Conv5
DeConv2	3	2	768/128	32/64	Conv4⊕DeConv1
DeConv3	3	2	384/64	64/128	Conv3⊕DeConv2
DeConv4	3	2	192/64	128/256	Conv2⊕DeConv3
DeConv5	3	1	128/64	256/256	Conv1⊕DeConv4
DeConv6	3	1	128/1	256/256	DeConv5

* K: kernel size; S: stride; Ch: channel; Res: resolution.

TABLE IX
Architecture of D^d_{S2R}, D^d_{En}

Name	K	S	Ch I/O	Res I/O	Input
Conv1	3	1	3/64 or 1/64	256/256	I^{rgb} or I^{d}
Conv2	3	2	64/64	256/128	Conv1
Conv3	3	1	64/64	128/128	Conv2
Conv4	3	2	64/128	128/64	Conv3
Conv5	3	1	128/128	64/64	Conv4
Conv6	3	2	128/256	64/32	Conv5
Conv7	3	1	256/512	32/32	Conv6
Conv8	3	2	512/512	32/16	Conv7
Conv9	3	1	512/1	16/16	Conv8

* K: kernel size; S: stride; Ch: channel; Res: resolution.

Acknowledgement

We want to express our gratitude to the researchers contributing to the reference datasets utilized in our study.

References

[1] F. Endres, J. Hess, J. Sturm, D. Cremers, and W. Burgard, "3-d mapping with an rgb-d camera," *IEEE transactions on robotics*, vol. 30, no. 1, pp. 177–187, 2013.

[2] E. Johns, S. Leutenegger, and A. J. Davison, "Deep learning a grasp function for grasping under gripper pose uncertainty," in *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2016, pp. 4461–4468.

[3] M. Shen, Y. Gu, N. Liu, and G.-Z. Yang, "Context-aware depth and pose estimation for bronchoscopic navigation," *IEEE Robotics and Automation Letters*, vol. 4, no. 2, pp. 732–739, 2019.

[4] Y. Guo, F. Deligianni, X. Gu, and G.-Z. Yang, "3-d canonical pose estimation and abnormal gait recognition with a single rgb-d camera," *IEEE Robotics and Automation Letters*, vol. 4, no. 4, pp. 3617–3624, Oct 2019.

[5] A. Atapour-Abarghouei and T. P. Breckon, "A comparative review of plausible hole filling strategies in the context of scene depth image completion," *Computers & Graphics*, vol. 72, pp. 39–58, 2018.

[6] G. Halmetschlager-Funek, M. Suchi, M. Kampel, and M. Vincze, "An empirical evaluation of ten depth cameras: Bias, precision, lateral noise, different lighting conditions and materials, and multiple sensor setups in indoor environments," *IEEE Robotics & Automation Magazine*, no. 99, pp. 1–1, 2018.

[7] T.-W. Hui, C. C. Loy, and X. Tang, "Depth map super-resolution by deep multi-scale guidance," in *European conference on computer vision*. Springer, 2016, pp. 353–369.

[8] X. Song, Y. Dai, and X. Qin, "Deep depth super-resolution: Learning depth super-resolution using deep convolutional neural network," in *Asian conference on computer vision*. Springer, 2016, pp. 360–376.

[9] A. Shrivastava, T. Pfister, O. Tuzel, J. Susskind, W. Wang, and R. Webb, "Learning from simulated and unsupervised images through adversarial training," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 2107–2116.

[10] A. Bulat, J. Yang, and G. Tzimiropoulos, "To learn image super-resolution, use a gan to learn how to do image degradation first," in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018, pp. 185–200.

[11] A. Handa, T. Whelan, J. McDonald, and A. J. Davison, "A benchmark for rgb-d visual odometry, 3d reconstruction and slam," in *2014 IEEE international conference on Robotics and automation (ICRA)*. IEEE, 2014, pp. 1524–1531.

[12] B. Planche, Z. Wu, K. Ma, S. Sun, S. Kluckner, O. Lehmann, T. Chen, A. Hutter, S. Zakharov, H. Kosch *et al.*, "Depthsynth: Real-time realistic synthetic data generation from cad models for 2.5 d recognition," in *2017 International Conference on 3D Vision (3DV)*. IEEE, 2017, pp. 1–10.

[13] M. Keller and A. Kolb, "Real-time simulation of time-of-flight sensors," *Simulation Modelling Practice and Theory*, vol. 17, no. 5, pp. 967–978, 2009.

[14] A. Atapour-Abarghouei, S. Akcay, G. P. de La Garanderie, and T. P. Breckon, "Generative adversarial framework for depth filling via wasserstein metric, cosine transform and domain transfer," *Pattern Recognition*, vol. 91, pp. 232–244, 2019.

[15] C. Sweeney, G. Izatt, and R. Tedrake, "A supervised approach to predicting noise in depth images," in *2019 International Conference on Robotics and Automation (ICRA)*. IEEE, 2019, pp. 796–802.

[16] W. He, Z. Xie, Y. Li, X. Wang, and W. Cai, "Synthesizing depth hand images with gans and style transfer for hand pose estimation," *Sensors*, vol. 19, no. 13, p. 2919, 2019.

[17] J. Kim, G. Jeon, and J. Jeong, "Joint-adaptive bilateral depth map upsampling," *Signal Processing: Image Communication*, vol. 29, no. 4, pp. 506–513, 2014.

[18] L. Zhao, H. Bai, A. Wang, Y. Zhao, and B. Zeng, "Two-stage filtering of compressed depth images with markov random field," *Signal Processing: Image Communication*, vol. 54, pp. 11–22, 2017.

[19] J. Yang, X. Ye, K. Li, C. Hou, and Y. Wang, "Color-guided depth recovery from rgb-d data using an adaptive autoregressive model," *IEEE transactions on image processing*, vol. 23, no. 8, pp. 3443–3458, 2014.

[20] B. Huhle, T. Schairer, P. Jenke, and W. Straßer, "Fusion of range and color images for denoising and resolution enhancement with a non-local filter," *Computer vision and image understanding*, vol. 114, no. 12, pp. 1336–1345, 2010.

[21] D. Ferstl, C. Reinbacher, R. Ranftl, M. Rüther, and H. Bischof, "Image guided depth upsampling using anisotropic total generalized variation," in *Proceedings of the IEEE International Conference on Computer Vision*, 2013, pp. 993–1000.

[22] W. Liu, X. Chen, J. Yang, and Q. Wu, "Robust color guided depth map restoration," *IEEE Transactions on Image Processing*, vol. 26, no. 1, pp. 315–327, 2017.

[23] X. Liu, D. Zhai, R. Chen, X. Ji, D. Zhao, and W. Gao, "Depth restoration from rgb-d data via joint adaptive regularization and thresholding on manifolds," *IEEE Transactions on Image Processing*, vol. 28, no. 3, pp. 1068–1079, 2018.

[24] J. Shen and S.-C. S. Cheung, "Layer depth denoising and completion for structured-light rgb-d cameras," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2013, pp. 1187–1194.

[25] G. Liu, F. A. Reda, K. J. Shih, T.-C. Wang, A. Tao, and B. Catanzaro, "Image inpainting for irregular holes using partial convolutions," in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018, pp. 85–100.

[26] J. Liu and X. Gong, "Guided depth enhancement via anisotropic diffusion," in *Pacific-Rim Conference on Multimedia*. Springer, 2013, pp. 408–417.

[27] S. Lu, X. Ren, and F. Liu, "Depth enhancement via low-rank matrix completion," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2014, pp. 3390–3397.

[28] Y. Zhang and T. Funkhouser, "Deep depth completion of a single rgb-d image," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 175–185.

[29] C. Dong, C. C. Loy, K. He, and X. Tang, "Image super-resolution using deep convolutional networks," *IEEE transactions on pattern analysis and machine intelligence*, vol. 38, no. 2, pp. 295–307, 2016.

[30] Y. Li, J.-B. Huang, N. Ahuja, and M.-H. Yang, "Deep joint image filtering," in *European Conference on Computer Vision*. Springer, 2016, pp. 154–169.

[31] L. Zhao, H. Bai, J. Liang, B. Zeng, A. Wang, and Y. Zhao, "Simultaneous color-depth super-resolution with conditional generative adversarial networks," *Pattern Recognition*, vol. 88, pp. 356–369, 2019.

[32] Y. Wen, B. Sheng, P. Li, W. Lin, and D. D. Feng, "Deep color guided coarse-to-fine convolutional network cascade for depth image super-resolution," *IEEE Transactions on Image Processing*, vol. 28, no. 2, pp. 994–1006, 2019.

[33] X. Song, Y. Dai, and X. Qin, "Deeply supervised depth map super-resolution as novel view synthesis," *IEEE Transactions on Circuits and Systems for Video Technology*, 2018.

[34] J. Chen, J. Chen, H. Chao, and M. Yang, "Image blind denoising with generative adversarial network based noise modeling," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 3155–3164.

[35] J. Jeon and S. Lee, "Reconstruction-based pairwise depth dataset for depth image enhancement using cnn," in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018, pp. 422–438.

[36] A. Dai, M. Nießner, M. Zollhöfer, S. Izadi, and C. Theobalt, "Bundlefusion: Real-time globally consistent 3d reconstruction using on-the-fly surface reintegration," *ACM Transactions on Graphics (ToG)*, vol. 36, no. 3, p. 24, 2017.

[37] J. Zeng, Y. Tong, Y. Huang, Q. Yan, W. Sun, J. Chen, and Y. Wang, "Deep surface normal estimation with hierarchical rgb-d fusion," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 6153–6162.

[38] M. Wang and W. Deng, "Deep visual domain adaptation: A survey," *Neurocomputing*, vol. 312, pp. 135–153, 2018.

[39] L. Zhang, "Transfer adaptation learning: A decade survey," *arXiv preprint arXiv:1903.04687*, 2019.

[40] J. Zhang, W. Li, P. Ogunbona, and D. Xu, "Recent advances in transfer learning for cross-dataset visual recognition: A problem-oriented perspective," *ACM Computing Surveys (CSUR)*, vol. 52, no. 1, pp. 1–38, 2019.

[41] W. Li, L. Chen, D. Xu, and L. Van Gool, "Visual recognition in rgb images and videos by learning from rgb-d data," *IEEE transactions on pattern analysis and machine intelligence*, vol. 40, no. 8, pp. 2030–2036, 2017.

[42] W. Li, Z. Xu, D. Xu, D. Dai, and L. Van Gool, "Domain generalization and adaptation using low rank exemplar svms," *IEEE transactions on pattern analysis and machine intelligence*, vol. 40, no. 5, pp. 1114–1127, 2017.

[43] W. Zhang, D. Xu, W. Ouyang, and W. Li, "Self-paced collaborative and adversarial network for unsupervised domain adaptation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2019.

[44] K. Bousmalis, A. Irpan, P. Wohlhart, Y. Bai, M. Kelcey, M. Kalakrishnan, L. Downs, J. Ibarz, P. Pastor, K. Konolige *et al.*, "Using simulation and domain adaptation to improve efficiency of deep robotic grasping," in *2018 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2018, pp. 4243–4250.

[45] M. Long, Y. Cao, J. Wang, and M. I. Jordan, "Learning transferable features with deep adaptation networks," *arXiv preprint arXiv:1502.02791*, 2015.

[46] Y. Ganin, E. Ustinova, H. Ajakan, P. Germain, H. Larochelle, F. Laviolette, M. Marchand, and V. Lempitsky, "Domain-adversarial training of neural networks," *The Journal of Machine Learning Research*, vol. 17, no. 1, pp. 2096–2030, 2016.

[47] Z. He and L. Zhang, "Multi-adversarial faster-rcnn for unrestricted object detection," in *Proceedings of the IEEE International Conference on Computer Vision*, 2019, pp. 6668–6677.

[48] F. Zhuang, X. Cheng, P. Luo, S. J. Pan, and Q. He, "Supervised representation learning: Transfer learning with deep autoencoders," in *Twenty-Fourth International Joint Conference on Artificial Intelligence*, 2015.

[49] C. Zheng, T.-J. Cham, and J. Cai, "T2net: Synthetic-to-realistic translation for solving single-image depth estimation tasks," in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018, pp. 767–783.

[50] S. James, P. Wohlhart, M. Kalakrishnan, D. Kalashnikov, A. Irpan, J. Ibarz, S. Levine, R. Hadsell, and K. Bousmalis, "Sim-to-real via sim-to-sim: Data-efficient robotic grasping via randomized-to-canonical adaptation networks," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 12 627–12 637.

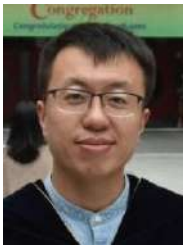
[51] J.-Y. Zhu, T. Park, P. Isola, and A. A. Efros, "Unpaired image-to-image translation using cycle-consistent adversarial networks," in *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 2223–2232.

[52] Z. Yi, H. Zhang, P. Tan, and M. Gong, "Dualgan: Unsupervised dual learning for image-to-image translation," in *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 2849–2857.

[53] Z. Li, H. Li, H. Han, G. Shi, J. Wang, and S. K. Zhou, "Encoding ct anatomy knowledge for unpaired chest x-ray image decomposi-

tion," in *International Conference on Medical Image Computing and Computer-Assisted Intervention*. Springer, 2019, pp. 275–283.
[54] S. Jeong, S. Kim, K. Park, and K. Sohn, "Learning to find unpaired cross-spectral correspondences," *IEEE Transactions on Image Processing*, vol. 28, no. 11, pp. 5394–5406, 2019.
[55] J. Hu, M. Ozay, Y. Zhang, and T. Okatani, "Revisiting single image depth estimation: Toward higher resolution maps with accurate object boundaries," *2019 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pp. 1043–1051, 2018.
[56] S. Zhao, H. Fu, M. Gong, and D. Tao, "Geometry-aware symmetric domain adaptation for monocular depth estimation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 9788–9798.
[57] F. Mahmood, R. Chen, and N. J. Durr, "Unsupervised reverse domain adaptation for synthetic medical images via adversarial training," *IEEE transactions on medical imaging*, vol. 37, no. 12, pp. 2572–2581, 2018.
[58] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.
[59] X. Mao, Q. Li, H. Xie, R. Y. Lau, Z. Wang, and S. Paul Smolley, "Least squares generative adversarial networks," in *Proceedings of the IEEE International Conference on Computer Vision*, 2017, pp. 2794–2802.
[60] W. Qiu and A. Yuille, "Unrealcv: Connecting computer vision to unreal engine," in *European Conference on Computer Vision*. Springer, 2016, pp. 909–916.
[61] A. Dai, A. X. Chang, M. Savva, M. Halber, T. Funkhouser, and M. Nießner, "Scannet: Richly-annotated 3d reconstructions of indoor scenes," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 5828–5839.
[62] N. Silberman, D. Hoiem, P. Kohli, and R. Fergus, "Indoor segmentation and support inference from rgbd images," in *European Conference on Computer Vision*. Springer, 2012, pp. 746–760.
[63] F. Ma, G. V. Cavalheiro, and S. Karaman, "Self-supervised sparse-to-dense: Self-supervised depth completion from lidar and monocular camera," in *2019 International Conference on Robotics and Automation (ICRA)*. IEEE, 2019, pp. 3288–3295.
[64] H. Zhang, T. Xu, H. Li, S. Zhang, X. Wang, X. Huang, and D. N. Metaxas, "Stackgan: Text to photo-realistic image synthesis with stacked generative adversarial networks," in *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 5907–5915.
[65] D. Liu, B. Wen, J. Jiao, X. Liu, Z. Wang, and T. S. Huang, "Connecting image denoising and high-level vision tasks via deep learning," *IEEE Transactions on Image Processing*, vol. 29, pp. 3695–3706, 2020.
[66] M. Heusel, H. Ramsauer, T. Unterthiner, B. Nessler, and S. Hochreiter, "Gans trained by a two time-scale update rule converge to a local nash equilibrium," in *Advances in Neural Information Processing Systems*, 2017, pp. 6626–6637.
[67] G. Petschnigg, R. Szeliski, M. Agrawala, M. Cohen, H. Hoppe, and K. Toyama, "Digital photography with flash and no-flash image pairs," *ACM transactions on graphics (TOG)*, vol. 23, no. 3, pp. 664–672, 2004.

Xiao Gu received the B.Eng. degree (Hons.) in electronic engineering from Fudan University, China, in 2018, and the M.Res. degree (Dean's Prize & Distinction) in medical robotics and image guided intervention from Imperial College London, U.K., in 2019, where he is currently pursuing the Ph.D. degree with the Hamlyn Centre. His research interests include biomedical signal processing, computer vision, transfer learning, and representation learning, especially in healthcare applications.



Yao Guo (S'14-M'18) received the B.S. and M.S. degrees in electrical engineering from Sun Yat-sen University, Guangzhou, China, in 2011 and 2014, respectively, and the Ph.D. degree in robotic vision from the City University of Hong Kong, Hong Kong, Chnia, in 2018. He is currently a Research Associate with the Hamlyn Centre for Robotic Surgery, Imperial College London, London. His research interests include robotic vision, pattern recognition, and machine learning for human motion analysis. He received the Best Conference Paper Award at ICMA2016.

Fani Deligianni holds a PhD in Medical Image Computing at Imperial College London, UK an MSc in Advanced Computing at Imperial College London, UK an MSc in Neuroscience at University College London, UK and a MEng (equivalent) in Electrical and Computer Engineering at Aristotle University, Greece. Currently, she is a Lecturer at School of Computing Science at Glasgow University. Her interests include medical image/neuroimage computing, statistical machine learning and bioinformatics as well as human motion analysis with wearable sensors and brain computer interfaces.

Guang-Zhong Yang (S'90–M'91–SM'08–F'11) is the Founding Dean of Institute of Medical Robotics, Shanghai Jiao Tong University. He used to be director and co-founder of the Hamlyn Centre for Robotic Surgery, Deputy Chairman of the Institute of Global Health Innovation, Imperial College London, UK. Professor Yang also holds a number of key academic positions at Imperial – he is Director and Founder of the Royal Society/Wolfson Medical Image Computing Laboratory, co-founder of the Wolfson Surgical Technology Laboratory, Chairman of the Centre for Pervasive Sensing. He is a Fellow of the Royal Academy of Engineering, fellow of IEEE, IET, AIMBE and a recipient of the Royal Society Research Merit Award and listed in The Times Eureka 'Top 100' in British Science.

Professor Yang's main research interests are in medical imaging, sensing and robotics. In imaging, he is credited for a number of novel MR phase contrast velocity imaging and computational modelling techniques that have transformed in vivo blood flow quantification and visualisation. These include the development of locally focused imaging combined with real-time navigator echoes for resolving respiratory motion for high-resolution coronary angiography, as well as MR dynamic flow pressure mapping for which he received the ISMRM I. I Rabi Award. He pioneered the concept of perceptual docking for robotic control, which represents a paradigm shift of learning and knowledge acquisition of motor and perceptual/cognitive behaviour for robotics, as well as the field of Body Sensor Network (BSN) for providing personalized wireless monitoring platforms that are pervasive, intelligent, and context-aware.